PCT

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification [6]: G06F 17/30	**A1**	**(11) International Publication Number:** WO 96/00946 **(43) International Publication Date:** 11 January 1996 (11.01.96)

(54) Title: DATA PRE-FETCH FOR SCRIPT-BASED MULTIMEDIA SYSTEMS
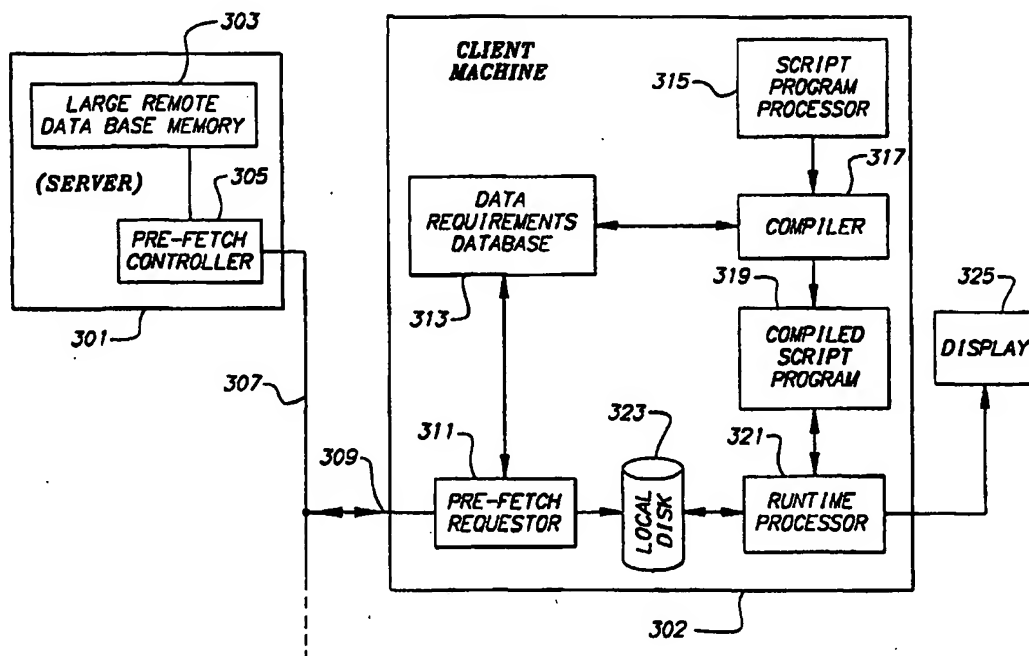
(57) Abstract

A multimedia architecture for pre-fetching data from a server provides limited-memory client machines with the ability to take advantage of a large remote database with relatively quick response time. A compiler program analyzes a given script to determine what data is needed by the current script and what potential scripts might be addressed from the current script and what server-stored data those scripts would need. That data is pre-fetched from the server upon commands from the compiler program in order to be quickly available for display at the client machine if requested by a user.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	GB	United Kingdom	MR	Mauritania
AU	Australia	GE	Georgia	MW	Malawi
BB	Barbados	GN	Guinea	NE	Niger
BE	Belgium	GR	Greece	NL	Netherlands
BF	Burkina Faso	HU	Hungary	NO	Norway
BG	Bulgaria	IE	Ireland	NZ	New Zealand
BJ	Benin	IT	Italy	PL	Poland
BR	Brazil	JP	Japan	PT	Portugal
BY	Belarus	KE	Kenya	RO	Romania
CA	Canada	KG	Kyrgystan	RU	Russian Federation
CF	Central African Republic	KP	Democratic People's Republic	SD	Sudan
CG	Congo		of Korea	SE	Sweden
CH	Switzerland	KR	Republic of Korea	SI	Slovenia
CI	Côte d'Ivoire	KZ	Kazakhstan	SK	Slovakia
CM	Cameroon	LI	Liechtenstein	SN	Senegal
CN	China	LK	Sri Lanka	TD	Chad
CS	Czechoslovakia	LU	Luxembourg	TG	Togo
CZ	Czech Republic	LV	Latvia	TJ	Tajikistan
DE	Germany	MC	Monaco	TT	Trinidad and Tobago
DK	Denmark	MD	Republic of Moldova	UA	Ukraine
ES	Spain	MG	Madagascar	US	United States of America
FI	Finland	ML	Mali	UZ	Uzbekistan
FR	France	MN	Mongolia	VN	Viet Nam
GA	Gabon				

DATA PRE-FETCH FOR SCRIPT-BASED MULTIMEDIA SYSTEMS

BACKGROUND OF THE INVENTION

This invention relates to multimedia systems, and more particularly to script-based client-server systems with remote storage of the main database. A main object of client-server systems is to provide limited-memory client machines (e.g. personal computers) with access to a voluminous database stored in a central server. The link between the server and the client machines is commonly accomplished by means of a single data line by employing time division multiplexing with the on-line client machines.

Generally, with such an arrangement, when a client machine requests server-stored data there results a passage of a significant "click-wait-watch" time period while the server proceeds to transmit the desired data to the client machines's local memory along a limited-bandwidth (i.e. relatively slow) information channel. With many multimedia applications, this slow response time renders the application significantly less useful (and in some cases unworkable).

This problem has been a major obstacle to efficient use of the client-server technique to advanced multimedia applications.

It is therefore an object of the instant invention to reduce the latency period for client receipt of requested information in client-server multimedia environments.

It is a further object of the instant invention to supply only the currently potentially-needed data to the client machine just in time upon request.

It is yet a further object of the instant invention to provide a script-analyzing compiler program that controls the pre-fetching of data in accordance with specific applications' script routines.

SUMMARY OF THE INVENTION

Applicant has developed a novel architecture for providing quick response time in a script-based multimedia client-server system. The client machines (e.g. PCs) employ a hierarchial multimedia script language which is used to describe a decision tree wherein each of the decision points, i.e. "nodes," have several other nodes to which they can proceed depending on the user's input. Once a particular node is chosen, multiple possible ensuing choices are made available to the user. The necessary data for these possible choices is pre-fetched automatically by a runtime processor that determines what data might be needed based on the user's current script node location as signaled to the runtime processor by the compiler. Just this potentially immediately-useful prefetched data is stored temporarily in a local "run time data requirement" memory (e.g. a special harddisk file). The compiler program is specially designed to analyze the script code to accomplish the suitable pre-fetching, so that a script

language application programmer can do the programming as

usual, and does not need to make any special pre-fetch requests

in the script code. There is no interference with the normal

operation of the scripting language. In this manner the click-

wait-watch times in the application program are minimized,

thus allowing limited machine memory PC users connected to the

large memory server to take advantage of advanced multimedia

applications that were previously practically realizable only

in conjunction with large memory machines.

BRIEF DESCRIPTION OF THE DRAWING

The reader's understanding of a practical implementa-

tion of a preferred embodiment of the instant invention will be

enhanced by reference to the following detailed description

taken in conjunction with the drawing figures, wherein:

FIGURE 1 depicts a client-server information exchange

system.

FIGURE 2 shows schematically a layered multimedia

script selection tree.

FIGURE 3 is a block diagram representation of a pre-

ferred architecture for implementing the method of the instant

invention.

DETAILED DESCRIPTION OF A PREFERRED EMBODIMENT

Referring now to Figure 1, therein depicted is a

client-server system for providing remote limited-memory client

machines 103-105, etc. (which are typically personal computers) with access to the extensive memory and database of a server 100 via communications channel 102. The communication channel can be a coaxial cable, a fibre optics cable, a microwave data link, or other similar link. This simple architecture is useful for some purposes, but in certain applications it is cumbersome, if not useless, due to the relatively long time delays (i.e. "click-wait-watch" time) involved in procuring the server-stored data.

In multimedia applications involving a "scripting" language it is desirable and often practically essential to keep all system delays to a minimum. Applications with long delays are simply not highly useful or marketable.

The instant invention addresses this problem in a novel way that can be better appreciated by reference to Figure 2. Therein depicted are various nodes of a scripting selection scheme. When a user selects an option, e.g. Node 1B, additional options become available to that user, e.g. Nodes 2F, 2G, and 2H. Each of these nodes in turn have further multiple options, and so on as shown.

Once an initial option is chosen, options for each ensuing layer are inherently contained in the script program. So, within the script program resides the information necessary to determine what immediate future selections are possible once the user has chosen a given node. The instant invention comprises taking advantage of this inherent information in the

script program by means of a specialized compiler program that
analyses the script program and determines automatically what
the potential immediate user options are at any given current-
ly-selected node. This information is used by the runtime
processor to immediately pre-fetch any _potentially_ needed data
from the server _prior_ to an actual request by the user for
particular data.

For example, once the user has chosen Node 2H in the
script selections, any data required from the server to fulfill
the requirements of _both_ Nodes 3J and 3K is automatically pre-
fetched from the server (in response to the commands of the
compiler) and locally-stored temporarily for potential near-
future use. In this regard, this specialized pre-fetch ar-
rangement is somewhat akin, in principle, to a "just in time"
parts delivery system for a manufacturing assembly line for
customized products. The object is to have the system work "on
the fly" as readily as possible without undue (or impossible)
local storage.   It should be noted that the technique
of the instant invention typically involves _more_ data being
transferred from the server to the client in a given time
period than would otherwise occur. This is due to the fact
that a significant portion of the data for all the potential
choices is transmitted even though only a fraction of it will
actually be used. Therefore, a practical implementation of the
instant invention can sometimes require an increased bandwidth
data channel between the server and clients beyond what would

-5-

otherwise be necessary. Nonetheless, at least from the user's standpoint, this is an acceptable and desirable tradeoff.

A preferred architecture for implementing the technique of the instant invention is depicted in Figure 3. Therein, server 301 comprising a large remote database memory 303 and a prefetch controller 305 is connected for two-way communication to a limited-memory client machine 302 (one of many such clients) via data channel 307 and data channel client connection 309. Pre-fetch requestor 311 receives input from data requirements database 313 corresponding to the potentially-needed "next nodes" of data choices (as described above with reference to Figure 2). The pre-fetch requestor then signals the server's pre-fetch controller to transmit the potentially-needed parts of next nodes of data immediately. Upon receipt of this data from the server, the prefetch requestor 311 directs the data to be temporarily stored in local disk 323.

The multimedia script is compiled by the compiler and generates two things: the compiled script program and the data requirements database (313). At a later time, the compiled script program is executed by a runtime processor (321) which executes the instructions from the compiled script, makes data requests, interacts with the user, and generates the desired output to display device 325.

The key to efficient operation of this system is the compiler program contained in compiler 317. This program

analyses the script program, at compile time, to determine what data is potentially needed next. This program controls the entire prefetch system. Those who are skilled in the art of drafting such programs will be able to prepare a suitable corresponding compiler program as a simple matter of programming for whatever particular type of client machine is used. The resultant codes will differ, but the basic concept of the invention will be equivalent.

The foregoing description refers to a preferred embodiment of Applicant's invention. The scope of the instant invention is not limited to this embodiment, but rather is defined by the appended claims and equivalents thereof.

What is claimed is:

1. In a script-based server-client system comprising a server machine with a large database memory and one or more client machines linked to said server by a data channel, a method for reducing user waiting times for an on-line client machine call-up of server-stored data comprising the steps of:

(A) determining from a script routine in said on-line client machine what particular server-stored data may be potentially presently-needed for display;

(B) pre-fetching said particular server-stored data by said on-line client machine from said server;

(C) temporarily storing said pre-fetched particular server-stored data in a local memory of said client machine; and

(D) upon command by said user, immediately displaying a selected portion of said pre-fetched data on a display device connected to said on-line client machine.

2. The method of Claim 1, wherein step (A) is accomplished by means of a compiler employing a specialized compiler program that analyzes said script routine.

3. The method of Claim 2 wherein an output of said compiler is used to control a data requirements database.

4. The method of Claim 3 wherein said data requirements database is used to control a pre-fetch requestor.

5.   The method of Claim 4 wherein said pre-fetch requestor supplies said pre-fetched particular server-stored data to said local memory comprising a hard disk memory.

6.   The method of Claim 4 wherein said pre-fetch requestor supplies said pre-fetched particular server-stored data to said local memory comprising a local RAM.

7.   The method of Claim 4 wherein said pre-fetch requestor supplies said pre-fetched particular server-stored data to said local memory comprising a local core memory.

8.   In a script-based client-server system comprising a server machine with a large database memory and one or more client machines linked to said server by a data channel, a apparatus for reducing user waiting times for an on-line client machine call-up of server-stored data comprising:

(A)   means for determining from a script routine in said on-line client machine what particular server-stored data may be potentially presently-needed for display;

(B)   means for pre-fetching said particular server-stored data by said on-line client machine from said server;

(C)   means for temporarily storing said pre-fetched particular server-stored data in a local memory of said client machine; and

(D)   means for immediately displaying a selected portion of said pre-fetched data, upon command by said user, on a display device connected to said on-line client machine.

-9-

9.   The apparatus of Claim 8, wherein said means for determining comprises compiler means employing a specialized compiler program that analyzes said script routine.

10. The apparatus of Claim 9 further comprising a data requirements database responsive to control signals from said compiler means.

11. The apparatus of Claim 10 further comprising pre-fetch requestor means responsive to control signals from said data requirements database.

12. The apparatus of Claim 11 including local memory means comprising a hard disk memory for receiving said pre-fetched particular server-stored data.

13. The apparatus of Claim 11 including local memory means comprising a local RAM memory for receiving said pre-fetched particular server-stored data.

14. The apparatus of Claim 11 including local memory means comprising a local core memory for receiving said pre-fetched particular server-stored data.
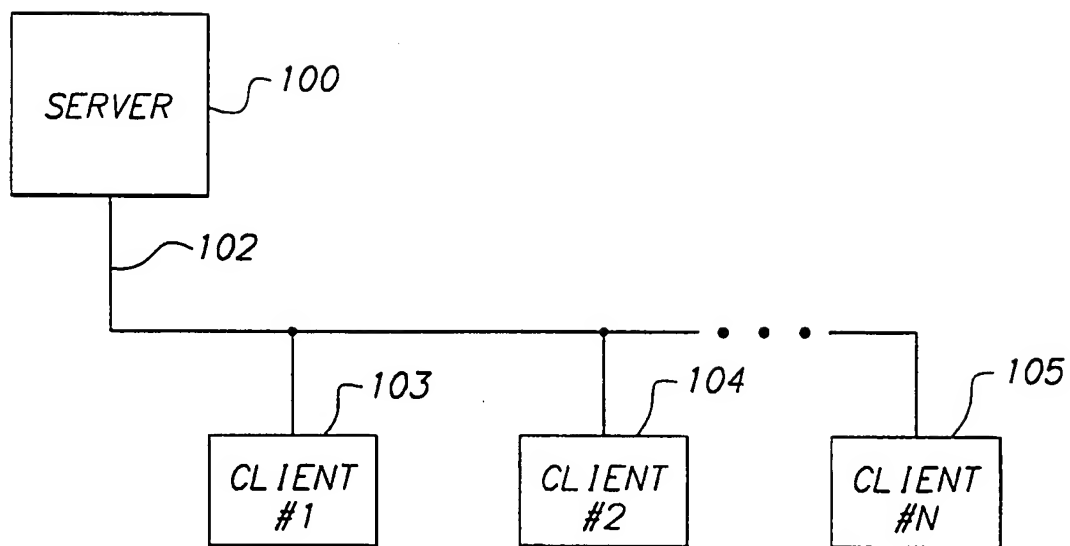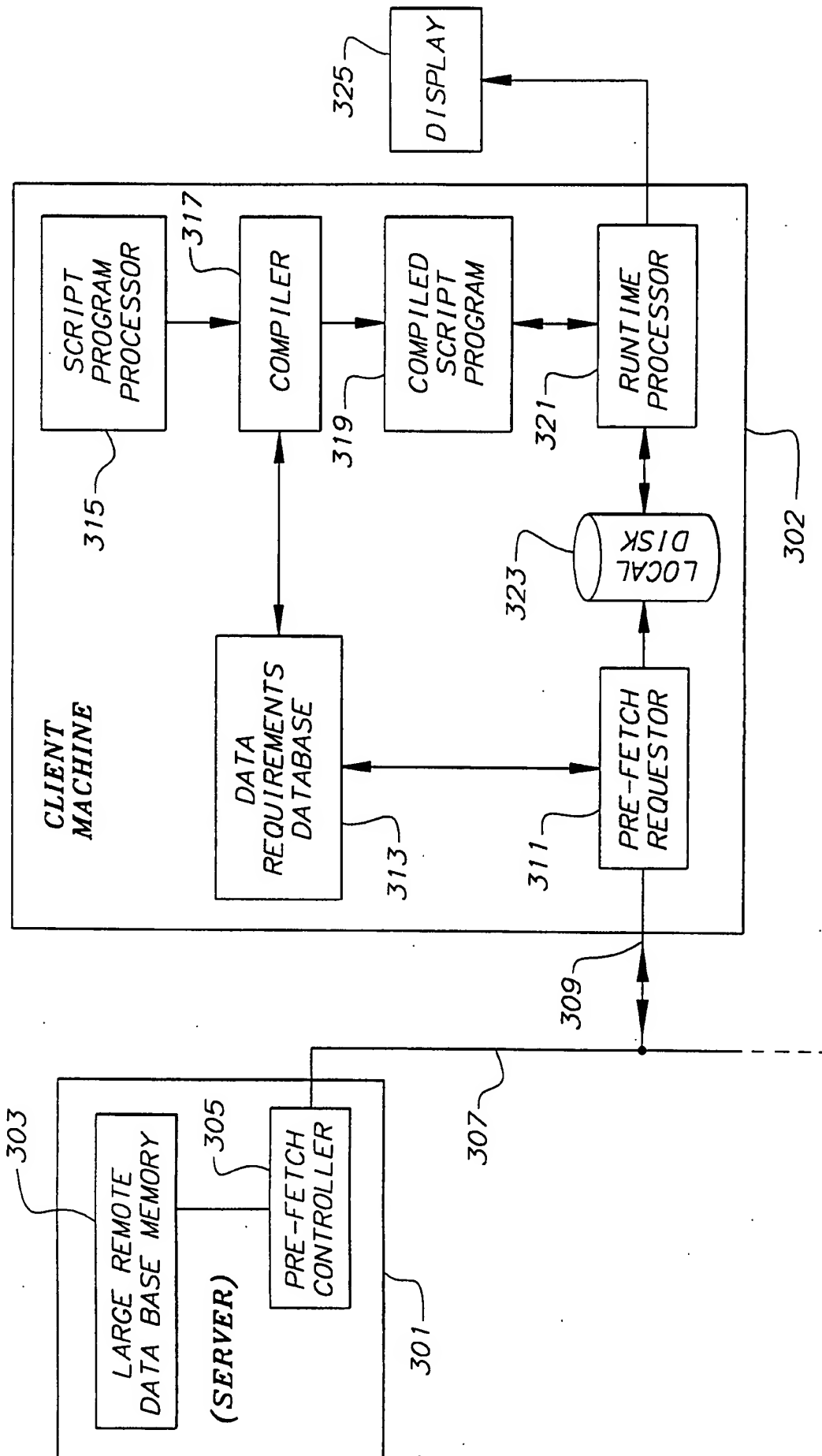
FIG. 1
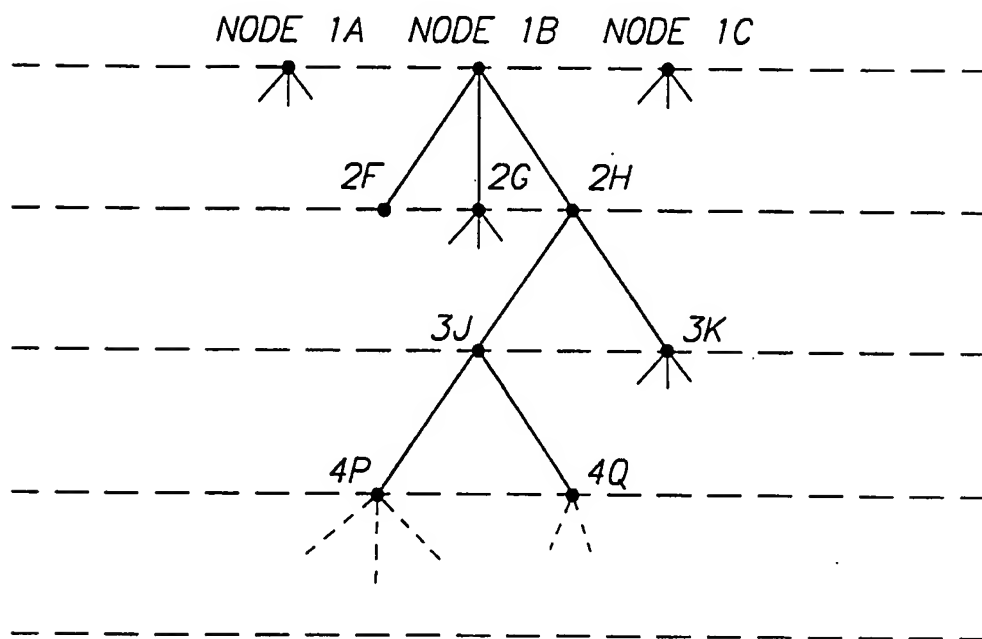
FIG. 2

1 / 2

FIG. 3

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6   G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6   G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	COMPUTER, vol. 26, no. 3, 1 March 1993 pages 44-53, XP 000364310 STAEHLI R ET AL 'CONSTRAINED-LATENCY STORAGE ACCESS' see the whole document ---	1,8
X	FOUNDATIONS OF DATA ORGANIZATION AND ALGORITHMS. 4TH INTERNATIONAL CONFERENCE. FODO '93 PROCEEDINGS, CHIGAGO, IL, USA, 13-15 OCT. 1993, ISBN 3-540-57301-1, 1993, BERLIN, GERMANY, SPRINGER-VERLAG, GERMANY, pages 1-18, MAIER D ET AL 'Storage system architectures for continuous media data' see the whole document ---	1,8

-/--

[X] Further documents are listed in the continuation of box C.   [ ] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
3 October 1995	19. 10. 95

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Katerbau, R

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON MULTIMEDIA COMPUTING AND SYSTEMS (CAT. NO.94TH0631-2), PROCEEDINGS OF IEEE INTERNATIONAL CONFERENCE ON MULTIMEDIA COMPUTING AND SYSTEMS, BOSTON, MA, USA, 15-19 MAY 1994, ISBN 0-8186-5530-5, 1994, LOS ALAMITOS, CA, USA, IEEE COMPUT. SOC. PRESS, USA, pages 224-233, RUBINE D ET AL 'Low-latency interaction through choice-points, buffering, and cuts in Tactus' see the whole document -----	1,8

1